Attorney's Docket No.: 42P15739

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

In re Application for:	
Francis X. McKeen	Examiner: Meonske, Tonia L.
Application No.: 10/644,399	Art Group: 2181
Filed: August 19, 2003	Conf. No.: 7924
For: **METHOD AND APPRATUS TO PROVIDE PROTECTION FROM A BUFFER OVERFLOW ATTACK**	

DECLARATION PURSUANT TO 37 C.F.R. §1.131

Mail Stop Amendment
Commissioner for Patents
P. O. 1450
Alexandria, VA 22313-1450

Dear Sir:

I, Francis X. McKeen, hereby declare that:

1. I am a citizen of the United States of America.
2. I currently reside at 10612 NW LeMans Ct. Portland, OR 97229.
3. I am currently an employee of Intel Corporation in Santa Clara, California.
4. I have been an employee of Intel Corporation since Oct 30, 1995.
5. My current title at Intel Corporation is Hardware Engineer.
6. I am the sole-inventor of the above-identified patent application.

7. I have reviewed U.S. Patent 6,996,677 issued to Lee et al. ("Lee"), which was filed on February 20, 2003. Lee claims priority from provisional patent application No. 60/429,839 filed on November 25, 2002. The Examiner cites Lee against the claims of the above-identified application.

8. The invention disclosed and claimed in the above-identified patent application was conceived in the United States of America at least as early as October 18, 2002, as evidenced by Intel Corporation Invention Disclosing Form (IDF) having ID #28002 (a copy of which is attached herein). This document was reduced to writing internally within Intel Corporation at least as early as the date on the document; i.e., October 18, 2002. The foils referenced by the IDF is a presentation entitled "LT Stack Protection," (a copy of which is attached herein). This document demonstrates conception of the claimed invention of the instant application. Although Revision 0.1 of the LT Stack Protection document indicates an August 28, 2007 date, as indicated in the attached screen print, the document was first created on March 9, 2001. Revision 1 of the LT Stack protection document was completed at least as early as the date on the date on the IDF document; i.e., October 18, 2002. Between at least October 2002 and its constructive reduction to practice by the filing of the above-captioned patent application on August 19, 2003, I directed various meetings with Intel's software and hardware design teams in a diligent effort to reduce the invention to practice. In addition, as a result of the meeting, I revised the LT Stack Protection document to provide Revision 0.2 of the LT Stack Protection document (a copy of which is attached herein). Revision 0.2 of the LT Stack Protection document was reduced to writing internally within Intel Corporation at least as early as the date on the document; i.e., July 2, 2003. Revision 0.2 of the LT Stack Protection document provides evidence of diligence between February 2003 and the constructive reduction to practice of the claimed invention of the instant application by the filing of the above captioned patent application on August 19, 2003. Therefore, the conception and diligence towards reduction to practice of the invention disclosed and claimed in the above-identified patent application occurred prior to the filing date of Lee.

9. The documents provided herewith are confidential. It is Intel Corporation's practice to maintain in secrecy all confidential documents. I believe that the documents have at all times prior to the filing date of the above-captioned application been maintained in a confidential manner.

I hereby declare that all statements made herein of my own knowledge are true and that all statements made on information and belief are believed to be true; and further that these statements were made with the knowledge that willful false statements and the like are punishable by fine or imprisonment, or both, under Section 1001 of Title 18 of the United States Code, and that such willful false statements may jeopardize the validity of the above-identified application or any patent issued thereon.

Respectfully submitted,

Dated: 1 Oct., 2007

Francis X. McKeen
Francis X. McKeen

Full Name:

Francis X. McKeen
Citizenship: United States of America
Residence: 10612 NW LeMans Ct.
Portland, OR 97229

~~I~~TEL INVENTION DISCLOSU~~RE~~

ATTORNEY-CLIENT PRIVILEGED COMMUNICATION
located at http://legal.intel.com/patent/index.htm

28002

DATE: October 18, 2002

MOBILE PLATFORMS/MPG/MPA

It is important to provide accurate and detailed information on this form. The information will be used to evaluate your invention for possible filing as a patent application. **Invention Disclosure forms MUST be sent electronically via email to your manager/supervisor who should then forward with their approval to our email account "invention disclosure submission."** If you have any questions, please call **8-264-0444.**

Last Name: McKeen	First Name: Francis (Frank)		M.I. X
Intel Phone Number:	Intel Fax Number:		Mailstop: CO5-166
E-mail address: frank.mckeen@intel.com			WWID: 10075788
Citizenship: USA	Are you a contractor?	Yes:	No: X
Home Address: 10612 NW LeMans Cr.			
City: Portland	State: OR	Zip: 97229	Country: USA
Corporate Level Group: MPG	Division:MPA		Subdivision: CASA
Supervisor: Krishnan Ravichandran	WWID: 10048707	M/S:RNB6-52	Phone #: 765-5308

Last Name:	First Name:		M.I.
Intel Phone Number:	Intel Fax Number:		Mailstop:
E-mail address:			WWID:
Citizenship:	Are you a contractor?	Yes:	No:
Home Address:			
City:	State:	Zip:	Country:
Corporate Level Group:	Division:		Subdivision:
Supervisor:	WWID:	M/S:	Phone #:

Last Name:	First Name:		M.I.
Intel Phone Number:	Intel Fax Number:		Mailstop:
E-mail address:			WWID:
Citizenship:	Are you a contractor?	Yes:	No:
Home Address:			
City:	State:	Zip:	Country:
Corporate Level Group:	Division:		Subdivision:
Supervisor:	WWID:	M/S:	Phone #:

(PROVIDE SAME INFORMATION AS ABOVE FOR EACH ADDITIONAL INVENTOR)

2. Title of Invention: A mechanism to protect from Stack Smashing Attacks on LT

3. What technology/product/process (code name) does your invention relate to (be specific if you can) La Grande Technology

4. Include several key words to describe the technology area of the invention in addition to # 3 above:
Buffer overflow; Stack smashing, virus attack

5. Stage of development (i.e. % complete, simulations done, test chips if any, etc.): Conceptual

6a. Has a description of your invention been (or planned to be) published outside of Intel: No.

If YES, was the manuscript submitted for pre-publication approval through the Author Incentive Program:

If YES, please identify the publication and the date published:

6b. Has your invention been used/sold or planned to be used/sold by Intel or others? No.

If YES, date it was sold or will be sold:

6c. Does this invention relate to technology that is or will be covered by a SIG (special interest group)/standard or specification? No.

If YES, name of SIG/standard/specification:

6d. If the Invention is embodied in a semiconductor device, actual or anticipated date of tapeout? Could be fall of 2003

6e. If the invention is software, actual or anticipated date of any beta tests outside Intel: Soon

7. Was the invention conceived or constructed in collaboration with anyone other than an Intel blue badge employee or in performance of a project involving entities other than Intel (e.g. government, other companies, universities or consortia)? NO: If YES, name of individual or entity:

8. Is this invention related to any other invention disclosure that you have recently submitted? If so, please give the title and inventors: No.

****

PLEASE READ AND FOLLOW THE DIRECTIONS ON HOW TO WRITE A DESCRIPTION OF YOUR INVENTION

Try to limit your description to 2-3 pages
Do NOT attach a presentation, white paper, or specification
ANSWER ALL OF THE QUESTIONS BELOW

Please provide a description of the Invention and include the following information:

1. **Describe in detail what the components of the invention are and how the invention works.**
   See foils inserted in email
2. **Describe advantage(s) of your invention over what is currently being done.**
   Currently there is no defense against buffer overflows and stack smashing attacks. A proposal to support a non LT version of dual stacks has been written up in an academic paper, Architectural Support for Defending Against Buffer Overflow Attacks, Xu, Kalbarczyk, Patel, Iyer, from the Center for Reliable Computing, University of Illinois, Urbana. There proposal is not the first to propose dual stacks. In this proposal there is no need to change legacy software to support the
3. **You MUST include at least one figure illustrating the invention. If the invention relates to software, include a flowchart or pseudo-code representation of the algorithm.**
   See the foilset
4. **Value of your invention to Intel (how will it be used?).**
   Allows Intel computers to stop spread of virus'.
5. **Explain how your invention is novel. If the technology itself is not new, explain what makes it different.**
   This invention allows virus protection of legacy code by use of the LT monitor. It allows LT to protect the LHS code integrity.
6. **Identify the closest or most pertinent prior art that you are aware of.**
   See paper reference above.
7. **Who is likely to want to use this invention or infringe the patent if one is obtained and how would infringement be detected? AMD, Microsoft**

HAVE YOUR SUPERVISOR READ AND FORWARD IT ELECTRONICALLY VIA E-MAIL TO "INVENTION DISCLOSURE SUBMISSION"

DATE: ________________ SUPERVISOR:

BY APPROVING, I (SUPERVISOR) ACKNOWLEDGE THAT I HAVE READ AND UNDERSTAND THIS DISCLOSURE, AND RECOMMEND THAT THE HONORARIUM BE PAID

RichEditWindow

LT Stack Protection

Frank McKeen

August 28, 2007

Rev 0.1

Intel Confidential

Agenda

MPG
Mobile Platforms Group

- Current LT value proposition
  - Solution gap
- Buffer overflow problem
- LT enhancement for stack attack mitigation

Intel Confidential

Security Concerns

MPG
Mobile Platforms Group

Activity	Risks/Concerns	Recommended Solution		
		2003	2004	2005-2006
Access data from enterprise	Confidential data intercepted	VPN/SSL	VPN/SSL	VPN/SSL
	Unauthorized access	PWD	TPM	TPM/LT
	Display/keyboard sniffing			LT
	Secure Transaction	VPN/SSL	VPN/SSL	LT
	Platform authentication	PWD	TPM	TPM/LT
E-commerce transaction	Secure Transaction	VPN/SSL	VPN/SSL	VPN/SSL/LT
	DRM	TRS	TRS	LT
	Display/keyboard sniffing			LT
	Platform authentication	PWD	TPM	TPM/LT
Email	Virus protection	Virus scan	Virus scan	LT may help
	Confid. email intercepted	IPSEC	IPSEC	IPSEC
	Stack Smash	Discipline	Discipline	Discipline
Use Wireless ad ess	Wireless data security	WEP	SSN	802.11i
	Platform authentication	PWD	TPM	TPM/LT
Notebook stolen	Data theft protection	PWD	Port token	Port token/LT
Exposed to internet	Virus protection	Virus scan	Virus scan	LT may help
	Stack Smash	Discipline	Discipline	Discipline
Use NB in publ	Password stolen	Educate	Educate	Educate
	Over shoulder reading	Educate	Educate	Educate

Intel Confidential

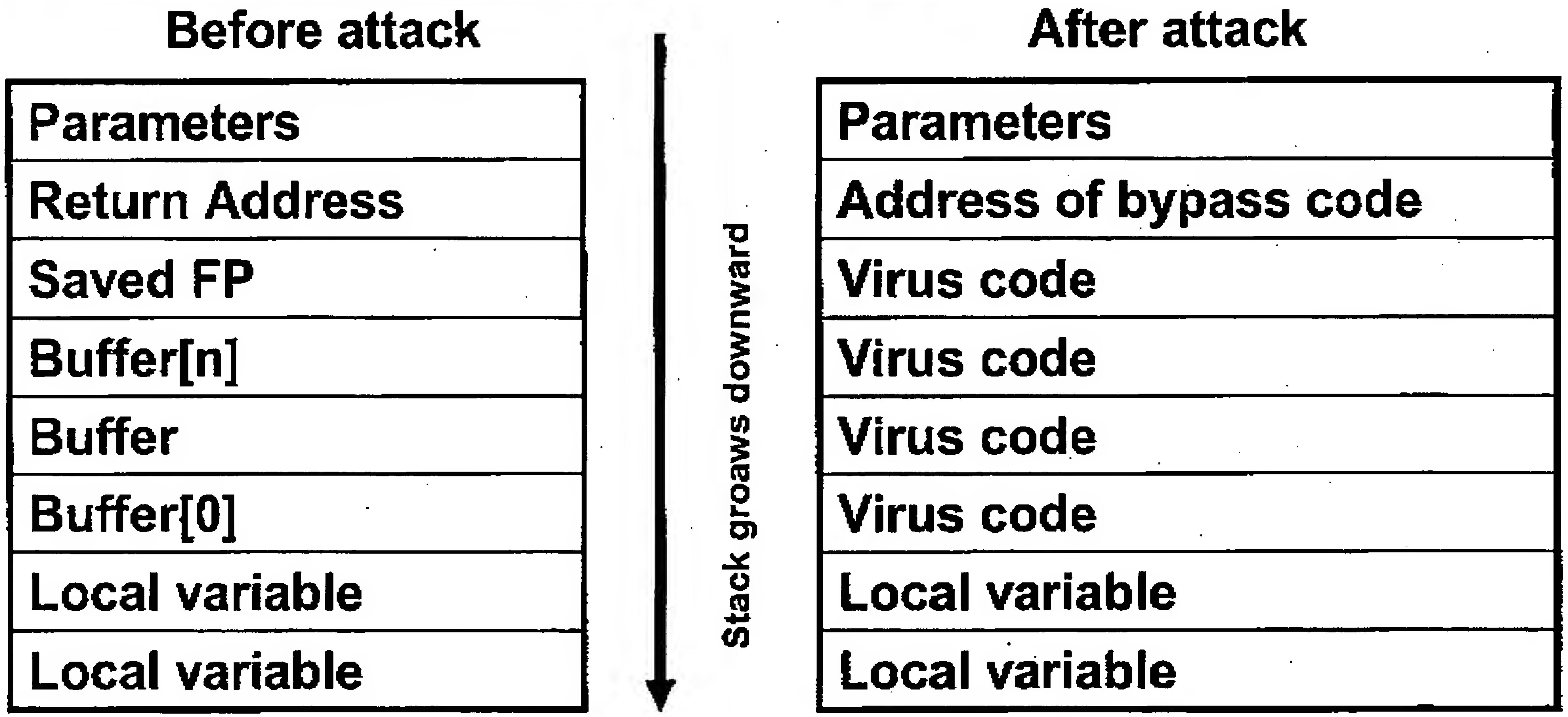
Stack Smash Attacks

- Stack Smashing attacks account for ~ 50% of security vulnerabilities reported.
- All major worms used stack smashing to bypass control of the machine
- LT does not currently solve stack smashing problem
  - LT would not protect against Code Red, Nimda, etc.
- LT impact on current internet security problems is limited to DRM and e-commerce

Enhance LT to reduce stack smashing

How smashing works

MPG
Mobile Platforms Group



Before attack
Parameters
Return Address
Saved FP
Buffer[n]
Buffer
Buffer[0]
Local variable
Local variable

Stack groaws downward

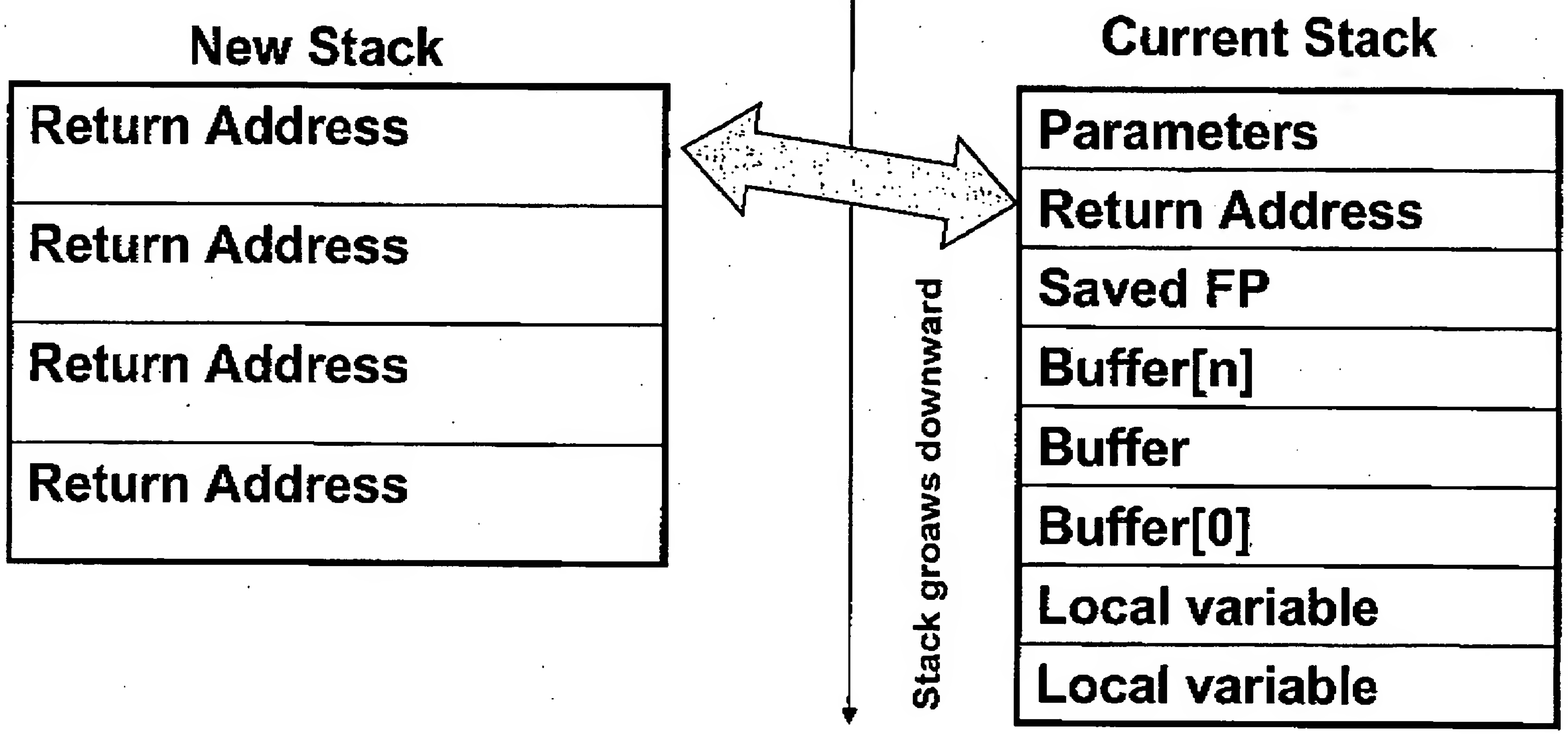
After attack
Parameters
Address of bypass code
Virus code
Virus code
Virus code
Virus code
Local variable
Local variable

Fill buffer and more to overwrite return address

- Data is written past the end of the buffer
  - Overwrites the return address
- Return address points to code which will redirect the program to new spot

Intel Confidential

Smashing protection



- Each Call deposits address in both stacks
- Each return checks that addresses match
- Failed matches are attacks.

Protection with LT

MPG

Mobile Platforms Group

- **Push the control stack into LT space where only the microcode and SVMM can touch it.**
  - Protects against other programs smashing both stacks
- **Microcode checks two stacks to validate correct address**
- **VMExit generated when the two values miscompare**
- **VMCS contains a bit which indicates the feature is enabled for a guest**
  - VMExit generated on loads to SP which relocate it.
  - Monitor maintains copy of the control stack.

Changes

- Microcode checks the values on both stacks
  - Can we experiment with patch?
- A second SP defined which points at the control stack. Each time the SP is loaded the CSP must be loaded.
- How do we determine live/dead for stacks?
  - Memory once used for stack could be kept active for a long time
  - All stacks are part of the memory image
  - Monitor keeps CSP as long as the stack is kept in memory.

MPG

Mobile Platforms Group

LT Stack Protection

Frank McKeen

July 2, 2003

Rev 0.2

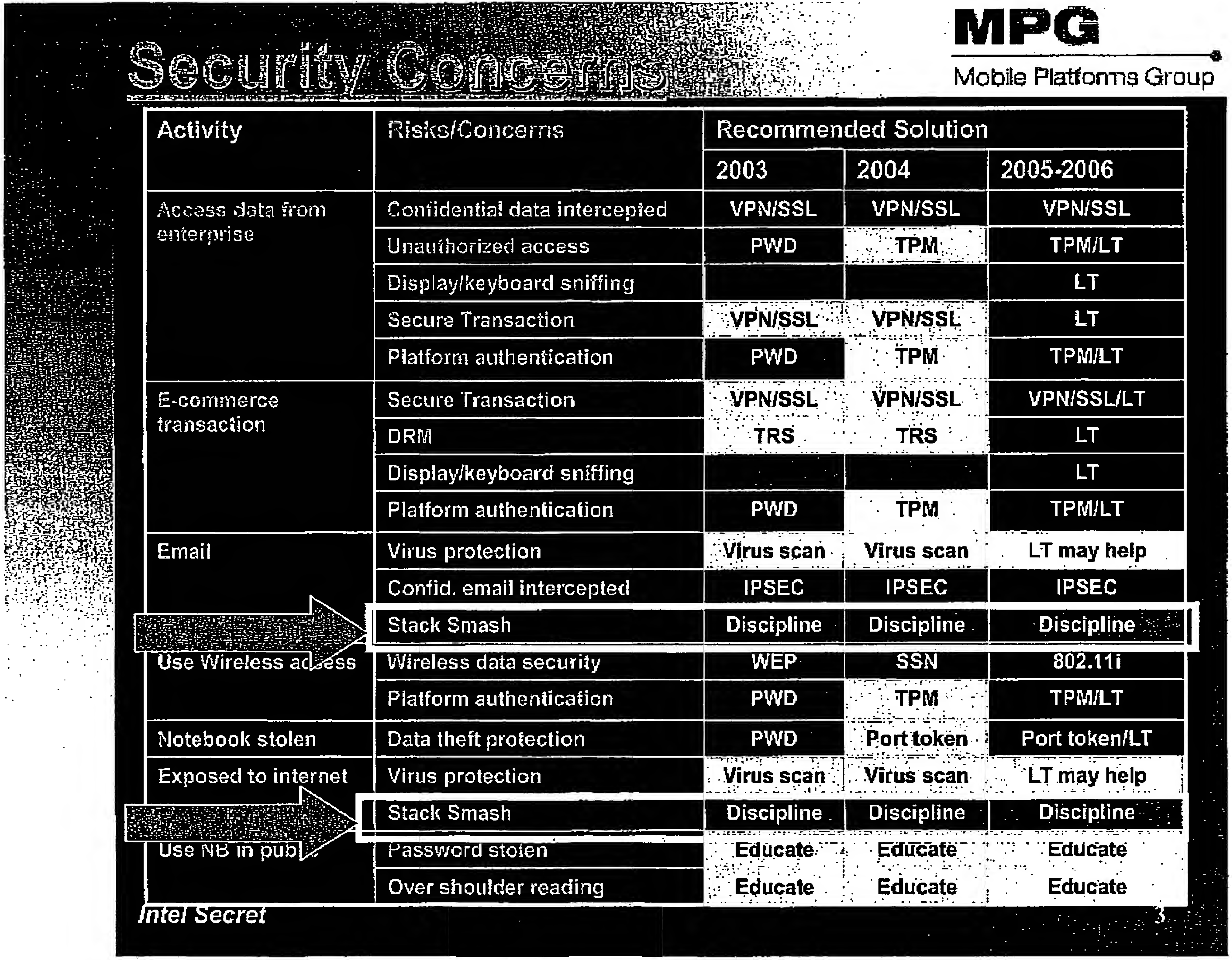
Intel Secret

Agenda

MPG
Mobile Platforms Group

- **Current LT value proposition**
  - – Solution gap
- **Buffer overflow problem**
- **LT enhancement for stack attack mitigation**

Intel Secret



Security Concerns

MPG
Mobile Platforms Group

Activity	Risks/Concerns	Recommended Solution		
		2003	2004	2005-2006
Access data from enterprise	Confidential data intercepted	VPN/SSL	VPN/SSL	VPN/SSL
	Unauthorized access	PWD	TPM	TPM/LT
	Display/keyboard sniffing			LT
	Secure Transaction	VPN/SSL	VPN/SSL	LT
	Platform authentication	PWD	TPM	TPM/LT
E-commerce transaction	Secure Transaction	VPN/SSL	VPN/SSL	VPN/SSL/LT
	DRM	TRS	TRS	LT
	Display/keyboard sniffing			LT
	Platform authentication	PWD	TPM	TPM/LT
Email	Virus protection	Virus scan	Virus scan	LT may help
	Confid. email intercepted	IPSEC	IPSEC	IPSEC
	Stack Smash	Discipline	Discipline	Discipline
Use Wireless access	Wireless data security	WEP	SSN	802.11i
	Platform authentication	PWD	TPM	TPM/LT
Notebook stolen	Data theft protection	PWD	Port token	Port token/LT
Exposed to internet	Virus protection	Virus scan	Virus scan	LT may help
	Stack Smash	Discipline	Discipline	Discipline
Use NB in pub	Password stolen	Educate	Educate	Educate
	Over shoulder reading	Educate	Educate	Educate

Intel Secret

3

Stack Smash Attacks

MPG
Mobile Platforms Group

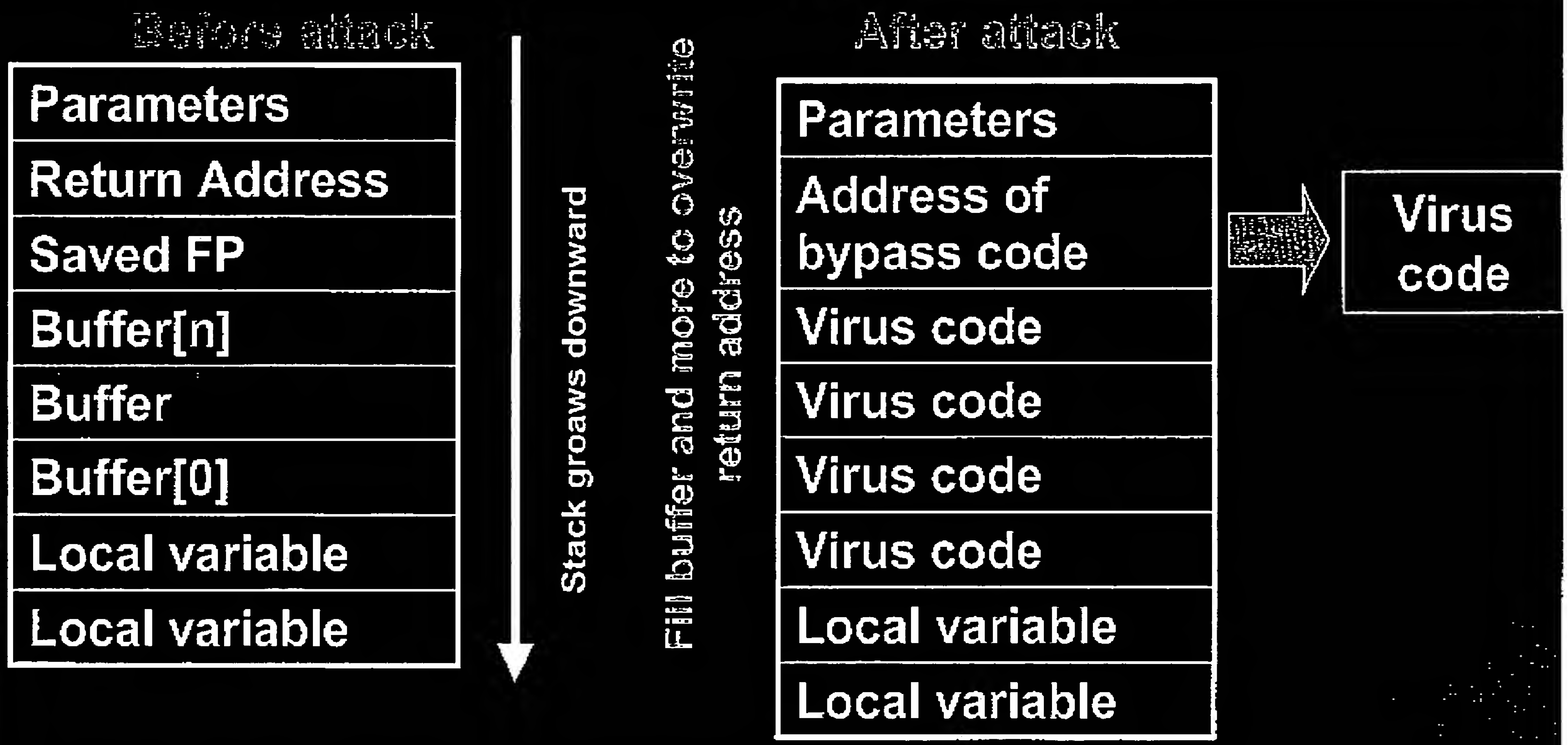
- Stack Smashing attacks account for ~ 50% of security vulnerabilities reported.
- All major worms used stack smashing to bypass control of the machine
- LT does not currently solve stack smashing problem
  - – LT would not protect against Code Red, Nimda, etc.
- LT impact on current internet security problems is limited to DRM and e-commerce
- Moving code to RHS does not mitigate threat

Enhance LT to reduce stack smashing

Intel Secret

MPG
Mobile Platforms Group

How smashing works

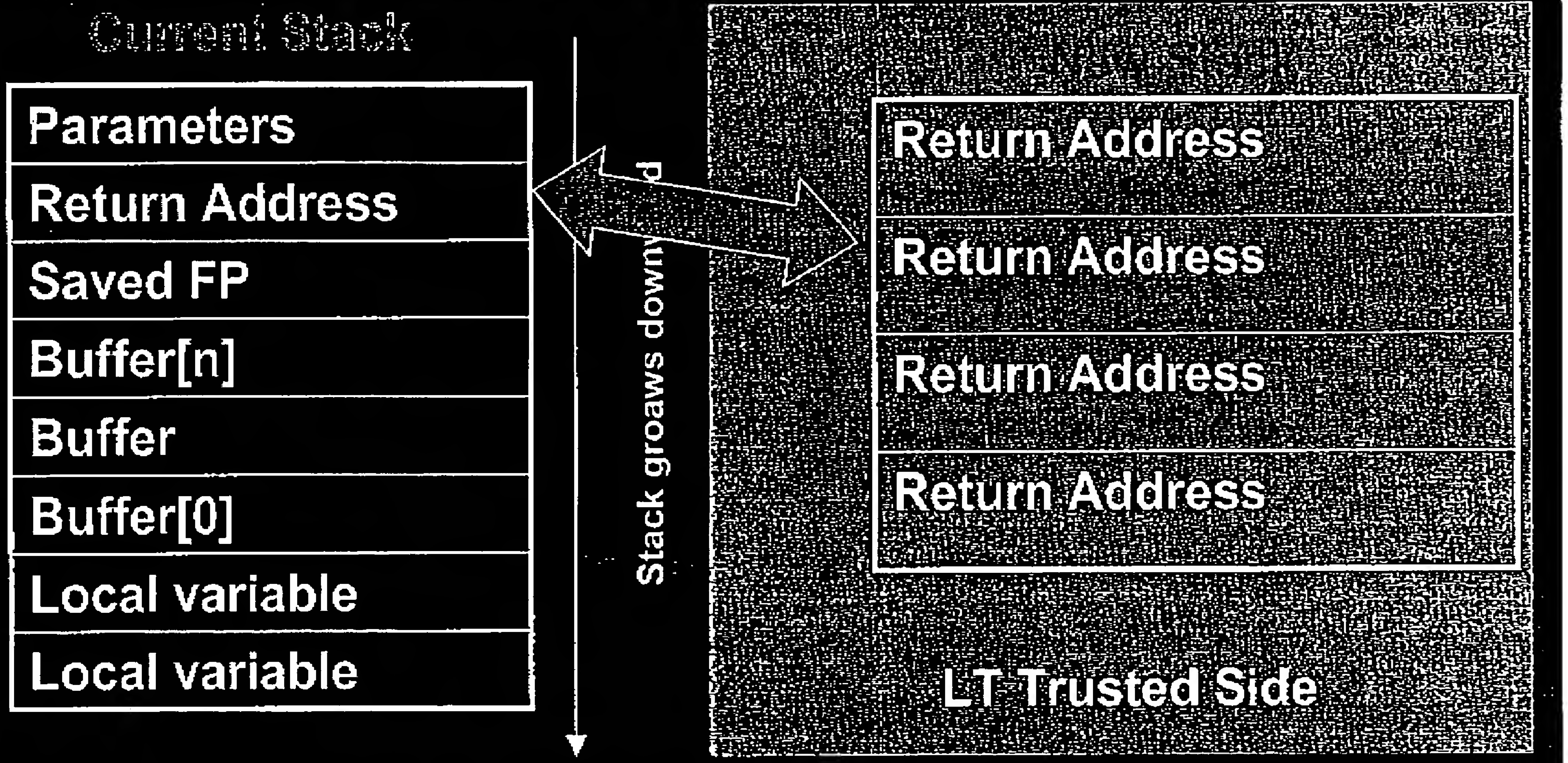


- Data is written past the end of the buffer
  - Overwrites the return address
- Return address points to code which will redirect the program to new spot
- Routine executes a return which gets the address of the virus code and passes control to the virus

Intel Secret

Smashing protection

MPG
Mobile Platforms Group



- Each Call deposits address in both stacks
- Each return checks that addresses match
- Failed matches are attacks.

Intel Secret

Protection with LT

MPG
Mobile Platforms Group

- Push the control stack into LT space where only the microcode and SVMM can touch it.
  - Protects against other programs smashing both stacks
- Microcode checks two stacks to validate correct address
- VMExit generated when the two values miscompare
- VMCS contains a bit which indicates the feature is enabled for a guest
  - VMExit generated on loads to SP which relocate it.
  - Monitor maintains copy of the control stack.

Intel Secret

Changes

- Microcode checks the values on both stacks
  - Can we experiment with patch?
- A second SP defined which points at the control stack. Each time the SP is loaded the CSP must be loaded.
- How do we determine live/dead for stacks?
  - Memory once used for stack could be kept active for a long time
  - All stacks are part of the memory image
  - Monitor keeps CSP as long as the stack is kept in memory.

Issues to resolve

MPG
Mobile Platforms Group

- SS, LSSS
- Task Switch
- ESP
- MOV SS
- Privilege level

Benefits

MPG
Mobile Platforms Group

- Improve integrity of the entire machine and avoid moving all software to the RHS.
- No changes to the current programming model for non monitor code
- Protects all legacy software
  - Reduces enabling effort substantially

Intel Secret

Backup

MPG
Mobile Platforms Group

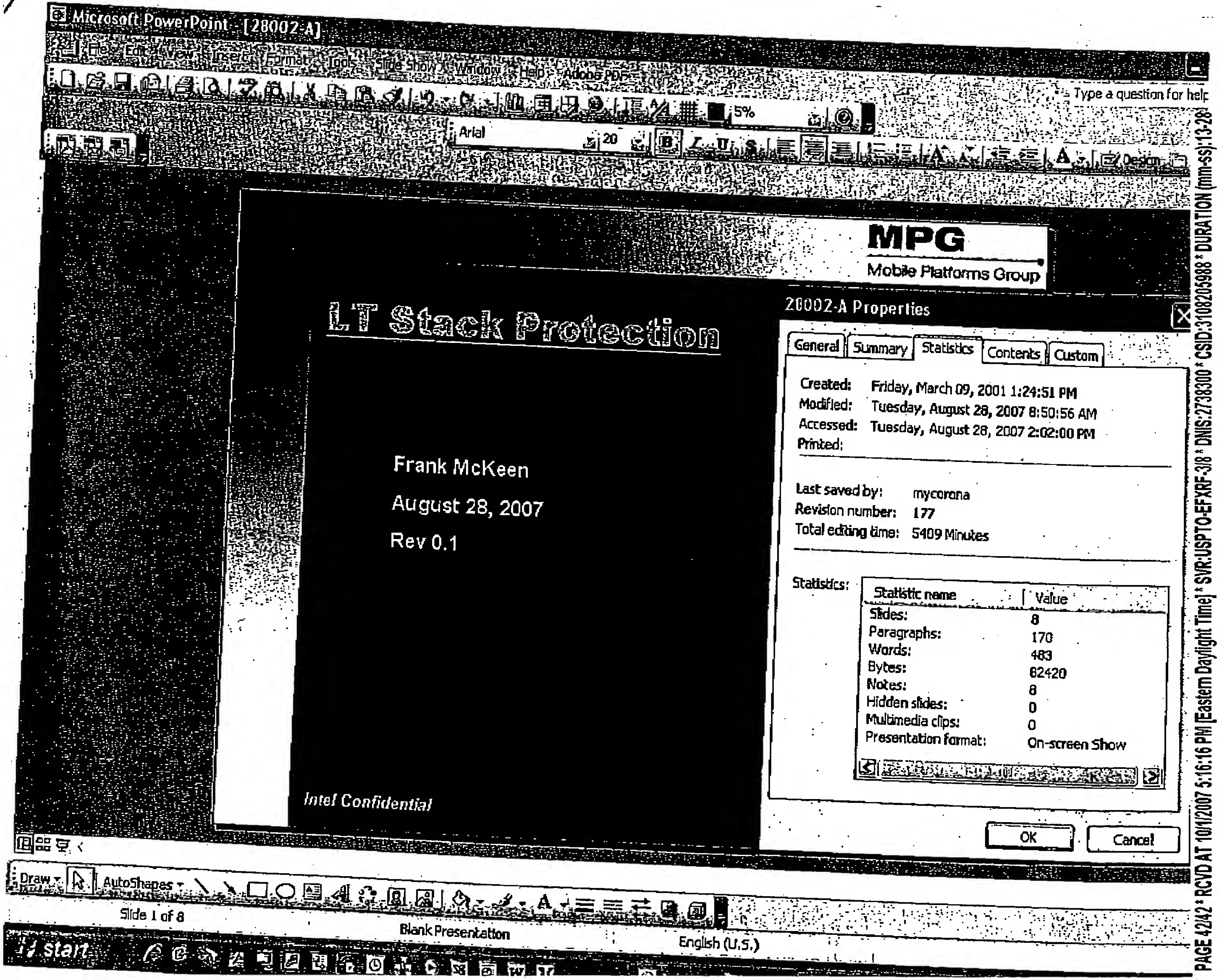
Intel Secret

MPG
Mobile Platforms Group

NonLT change

- The CSP must be defined architecturally as a register
  - Can't be accessed by current instruction definitions.
  - Use new instruction which sends data to CSP
- New exception should be defined to handle the mismatch case.
  - Not absolutely necessary but mainly a performance enhancement. Could use current event.
- Microcode change to patch the call and ret.
- Potentially faster time to market?

Intel Secret



MPG
Mobile Platforms Group

LT Stack Protection

Frank McKeen

August 28, 2007

Rev 0.1

Intel Confidential

28002-A Properties

Created: Friday, March 09, 2001 1:24:51 PM
Modified: Tuesday, August 28, 2007 8:50:56 AM
Accessed: Tuesday, August 28, 2007 2:02:00 PM
Printed:

Last saved by: mycorona
Revision number: 177
Total editing time: 5409 Minutes

Statistics:

Statistic name	Value
Slides:	8
Paragraphs:	170
Words:	483
Bytes:	82420
Notes:	8
Hidden slides:	0
Multimedia clips:	0
Presentation format:	On-screen Show